Annotating Relations in Scientific Articles

Adam Meyers[†], Giancarlo Lee[‡], Angus Grieve-Smith[†], Yifan He[†], Harriet Taber[†]

New York University[†], Cornell University[‡]
meyers@cs.nyu.edu, gcl58@cornell.edu, angus@cs.nyu.edu, yhe@cs.nyu.edu, htaber@cs.nyu.edu

Abstract

Relations (ABBREVIATE, EXEMPLIFY, ORIGINATE, REL_WORK, OPINION) between entities (citations, jargon, people, organizations) are annotated for PubMed scientific articles. We discuss our specifications, pre-processing and evaluation.

Keywords: Corpus Annotation, Relation Extraction, Technical Document Processing

1. Introduction

This paper reports on the annotation of relations between entities in scientific articles from the PubMed Central corpus (`http://www.ncbi.nlm.nih.gov/pmc/`). The development of these relations draws on previous work ranging from (Teufel et al., 2009; Athar, 2011) to Automatic Content Extraction (`http://projects.ldc.upenn.edu/ace/docs/EnglishRDCV4-3-2.PDF`). Examples of our relations are provided below as figure 1.

The **ABBREVIATE** relations connect two instances of terminology (terms or jargon) with a coreference relation. The **EXEMPLIFY** relation establishes that one term is a subtype of another (Hearst, 1992). The **ORIGINATE** relation establishes that a person/organization/cited document is the source (discoverer, manufacturer or supplier) of some object realized as a term. **RELATED_WORK** relations connect two documents, either cited documents or self-references to the current paper using *we*, *this research* and similar terms – we notate the self-reference case as ThisArticle. Subtypes of **RELATED_WORK** are distinguished by whether the documents are in contrast or corroborate, or whether one is based on the other. An **OPINION** relation indicates that the authors depict a citation or instance of a term as: practical, significant, standard, positive or negative. (See `https://sites.google.com/a/nyu.edu/specs/annotation-specifications`). For OPINION relations, A1 is implicitly the authors of the article.[1]

We use the following notation. Names of relations are in all capital letters, but can be omitted when discussing sub-types, e.g., PRACTICAL refers to OPINION:PRACTICAL. Arguments of a relation are in bold with an argument number (A1 or A2) as a subscript. Multiple instances of A1 or A2, indicate that multiple relations are found in a sentence. A signal indicating that a relation occurs is enclosed in a box if lexical, and listed in square brackets if grammatical. We use square brackets and numbers (IEEE style) to represent citations that are the object of this study. In contrast, we cite work related to this paper using APA style (last names and years).

We will now define the subtypes of relations. For ABBREVIATE, we distinguish the *ALIAS* subtype from other instances. All *ABBREVIATE* relations represent cases when A1 and A2 are potential references to the same concept. For normal *ABBREVIATE* cases, A1 is a full form and A2 is a shorter form used in the interest of brevity. The *ALIAS* subtype is appropriate when the A1 and A2 are alternatives with equal standing, possibly used in different contexts. In example 5, a chemical name and a chemical formula are both representations of the same substance. Yet neither form really abbreviates the other: they have almost the same number of characters; and their usage reflects different functions and/or emphasizes different properties (status as a salt vs combination of elements forming a molecule). *ALIAS* is a subtype of *ABBREVIATE* rather than a separate relation because +/-ALIAS relations have similar functions and can be difficult to differentiate in some cases. The subtypes of *ORIGINATE* reflect a degree of origination ranging from *DISCOVER* representing authorship, invention and discovery (example 10,11), to *SUPPLY* representing that the A1 merely supplies (sells, distributes) the A2 (example 13). *MANUFACTURE* is a middle ground: the A1 creates instances of, but may not have invented the A2 (example 12). As with *ALIAS*, the subtypes of *ORIGINATE* convey different, but related concepts, ones which are not always easy to differentiate. The subtypes of REL_WORK state how the A1 and A2 are related: CORROBORATE suggests that (part of) A1 corroborates with (part of) A2 (and the reverse may be true as well); CONTRAST suggests that A1 and A2 differ; BETTER_THAN is like CONTRAST, except that the difference suggests an advantage of A1 (in this limited respect) compared to A2; and BASED_ON suggests that A1 bases some important idea on A2. Subtypes of OPINION, state the nature of an opinion or assumption about A2. PRACTICAL (the most common opinion) marks any term or citation that is used for something or noted to be useful. STANDARD refers to terms/citations that represent something that has become standard (this classification overrides PRACTICAL as many standard items are also practical). SIGNIFICANT

[1] An A1 other than the authors is possible in theory. However, we found very few instances of this, and in each such case the opinion was shared by the authors, e.g., in the following example, the significance may be attributed to [4], but the authors only site this because they agree with the opinion: *The corresponding* **Tyr135**$_{A2}$ *of S. cerevisiae Spo11 is essential for recombination . . . [4]* {*OPINION:SIGNIFICANT*}.

1. **Netherlands Vaccine Institute**$_{A1}$ (**NVI**$_{A2}$) {ABBREVIATE} [PARENTHESES]
2. **highly pathogenic avian influenza virus**$_{A1}$ (**HPAIV**$_{A2}$){ABBREVIATE} [PARENTHESES]
3. **third variable loop**$_{ARG1}$ (**V3**$_{ARG2}$) {ABBREVIATE} [PARENTHESES]
4. **Bayesian Dirichlet metric**$_{ARG1}$ or **BDe**$_{ARG2}$ {ABBREVIATE}
5. **Silver behenate**$_{A1}$ (**CH3-(CH2)20-COOAg**$_{A2}$){ABBREVIATE:ALIAS} [PARENTHESES]
6. **housekeeping gene 36B4**$_{A1}$ (**acidic ribosomal phosphoprotein P0**$_{A2}$){ABBREVIATE:ALIAS} [PARENTHESES]
7. **Cytokines**$_{A2}$, for instance interferon **(IFN)-**$_{A1}$ ... {EXEMPLIFY}
8. This enabled certain **proteins**$_{A2}$ such as **insulin**$_{A1}$, **growth hormone**$_{A1}$, or **plasminogen activator**$_{A1}$ to be ... {3 EXEMPLIFY relations}
9. ... in the **I** κ **B protein**$_{A1}$, an **inhibitor of NF-** κ **B**$_{A2}$ {EXEMPLIFY} [APPOSITION]
10. **Benjamini**$_{A1}$**-Hochberg**$_{A1}$ **false discovery rate**$_{A2}$ {ORIGINATE:DISCOVER} [NOUN_MOD]
11. The **carotid artery ligation model**$_{A2}$ was described previously [**12**] {ORIGINATE:DISCOVER} [CITATION]
12. **Eagle's minimum essential media**$_{A2}$ (**ATCC**$_{A1}$) {ORIGINATE:MANUFACTURE} [PARENTHESES]
13. **DOPG**$_{A2}$ was obtained from **Avanti Polar Lipids**$_{A1}$. {ORIGINATE:Supply}
14. **These findings**$_{A1}$ are endorsed by the literature [**1**$_{A2}$] {REL_WORK:CORROBORATE}
15. Both FluA and FluB viruses have a common origin [**1**$_{A1}$]. Thus, it is not unexpected that the aa residues of PA are conserved between FluA and FluB [**25**$_{A2}$]. {REL_WORK:CORROBORATE}
16. Some species shed large amounts of virus, yet exhibit few detrimental effects [**3**$_{A1}$]. On the other hand, various species of swans (Cygnus spp.) and the wood duck (Aix sponsa), show 100% mortality within days of inoculation with HPAIV (H5N1) [**4**$_{A2}$]. {REL_WORK:CONTRAST}
17. a **necrotrophic effector system**$_{A1}$ that is an exciting contrast to the **biotrophic effector models**$_{A2}$ that have been intensively studied ... {REL_WORK:CONTRAST}
18. **Bayesian networks**$_{A1}$ hold a considerable advantage over **pairwise association tests**$_{A2}$. {REL_WORK:BETTER_THAN}
19. Ion exchange was used for the purification of BG220-reactive material as previously described [**20**$_{A2}$].{REL_WORK:BASED_ON} [ThisArticle is implied by passive *used*]
20. We ..., using an algorithm proposed by Friedman and Koller [**52**$_{A2}$]. {OPINION:PRACTICAL}
21. The **gene proteins**$_{A2}$ used in this experiment were also used in [2].{OPINION:PRACTICAL}
22. **Anaerobic SBs**$_{A2}$ are an emerging area of research and development {OPINION:SIGNIFICANT}
23. ... were derived by standard procedures as described in [**46**$_{A2}$]. {OPINION:STANDARD}
24. **Eagle's minimum essential media**$_{A2}$ {OPINION:STANDARD} [POSSESSIVE]
25. **Bayesian networks**$_{A2}$ {OPINION:STANDARD} [MORPHOLOGY]
26. A recent study using complementary DNA tags as adaptors to immobilize peptide:MHC on the slides is innovative **[16]**$_{A2}$. {OPINION:POSITIVE}
27. The treatments recommended by the World Health Organization for advanced HAT, however, **melarsoprol**$_{A2}$ and **eflornithine**$_{A2}$, are entirely inadequate. {OPINION:NEGATIVE} [2 relations]

Table 1: Examples of Annotated Relations

marks active areas of research or items worthy of study (e.g., potentially PRACTICAL in the future). POSITIVE and NEGATIVE are reserved for unambiguous statements of positive and negative opinion (examples 26 and 27), which we have found to be rare in the technical text that we have annotated. Our annotation guidelines attempted to account for apparent discrepancies between these subclasses, stating specifically how to deal with cases where multiple subtypes would seem to apply (e.g., POSITIVE is used sparingly, due to the more specific classes like PRACTICAL and SIGNIFICANT; STANDARD overrides PRACTICAL as it is more specific, etc.) We have found that OPINION:NEGATIVE almost never occurs in the documents that we annotated. Authors tend to compare particular features and evoke RELATED_WORK relations (CONTRAST or BETTER_THAN). Also, cited work was sometimes the basis of the current research (BASED_ON) and the author often stated improvements that this work made over the original research – however, it is not the case that the compared work was bad, merely that the new work was better. In contrast, work an author outright disliked was very rarely cited at all. We suspect that this may reflect

the sociology of the fields that we annotated (natural science) and that perhaps more volatile language appears in other technical literatures, e.g., the linguistics literature of the 1970s featured in (Harris, 1995).

2. Motivation

Researchers in the field of technology forecasting currently use citation analysis (Daim et al., 2006) to identify trends in technology and make future predictions. We believe that the identification of the right set of entities and relations could improve these techniques. Current citation graphs represent which documents cite which other documents and which pairs of documents are cited together. Changing patterns in these graphs over time are used to analyze trends in technology. Systems that automatically extract the relations we annotate could help fine tune these relations into meaningful patterns by: identifying that documents are cited as being practical, standard or significant (similar to (Teufel et al., 2009; Athar, 2011)); clustering or separating sets of documents reputed to corroborate or contrast (Meyers, 2013); or that one document is based on another. Our goal is to annotate a sufficient number of high quality instances of these relations, so that systems based on this annotation can be built. Following the conventional research paradigm in computational linguistics, our annotation guidelines defines a task for system developers and our annotated data will be used by system developers for evaluation and training purposes.

Terms, like citations, can be contrasted, corroborated, deemed practical/significant/standard. Additionally, only important terms and organizations tend to be abbreviated or aliased in other ways, or used in exemplify relations (hyponyms).[2] Furthermore, tracking the origination of documents and terms (theories, inventions) by originators (authors, inventors, suppliers), realized as people names and organizations, has also proved fruitful for technology forecasting. In addition, some of our relations provide the means for establishing coreference (ABBREVIATION) and similar relations such as hyponymy (EXEMPLIFY) and metonymy (using ORIGINATE, e.g., substituting *Freud* for Freud's theories.)

In related work, we have found that relations involving citations, technical terms, organizations, people and citations are important to technology forecasting that goes beyond pure citation analysis(Babko-Malaya et al., 2013b; Babko-Malaya et al., 2013a; Thomas et al., 2013). We have used our relations in systems for identifying communities of practice (e.g., OPINION) and debates (e.g., REL_WORK:CONTRAST relations) within those communities; and for determining when emerging fields have developed practical applications (e.g., OPINION:PRACTICAL).

[2]While our automatic abbreviation pre-processors are highly accurate (94%), annotation has shown that there is room for improvement on recall (58%), tested on 10 documents. In particular, while most instances of abbreviation occur in a couple of narrow syntactic environments: tables and inside/outside of parentheses. Other instances of abbreviation are linked by keywords (e.g., *henceforth, is denoted as, ...*) and not all mappings between full form and abbreviation are easily predicted (figure refall-examples, example 3). So systems can benefit from annotated data.

3. Signals and Arguments

For each annotated relation, we mark: (i) the arguments (A1 and A2) of the relation and (ii) a lexical item or grammatical construction type that the annotator identifies as the signal that a given relation holds. Together these items are the evidence in the text that the relation holds. While the arguments are the items that are "related", the signal is the trigger that indicates the relation holds. By requiring signals, we constrain the annotated relations to those that are learnable via machine learning (ML) methods, and we provide specific terms for ML to use for relation prediction. This approach is similar to that of semantic role labeling (Baker et al., 1998; Palmer et al., 2005; Meyers et al., 2004) and discourse argument (Miltsakaki et al., 2004) annotation and different from ACE's approach which allows annotators the leeway to annotate relations in sentences, provided that a reasonable interpretation would suggest that the relation holds. While our method was originally difficult to codify, we were ultimately able to achieve accurate annotation, as this paper shows (section 4.).

A signal can be either: (a) a word or word sequence; or (b) a grammatical signal from a finite list of possibilities. Lexical signals can be predicates (verb, noun, adjective), modifiers (adjective, adverb), prepositions or discourse connectives. A signal should "signal" that a relation exists and "link" the arguments. The arguments can be conjuncts of a conjunction signal (example 4), a prepositional object or modifiee of a preposition signal (example 8), argument of a verb (example 13), or clauses linked by a discourse connective signal (example 15). Adjectives can also be signals for the nouns they modify (example 23). In addition to these direct links, we allow some indirect links, we allow constituents that are "closely linked" to arguments/modifiees of the signals. For example, if an NP contains a signal adjective, we allow arguments that are linked to this NP by apposition or predication (example 22). Also, we assume that citations can be parenthetically associated with whole sentences (example 16), subjects of those sentences (example 11) or other constituents (example 14) and we look for signals that are associated with these constituents, treating them essentially as proxies for the citation. Grammatical signals include: *PARENTHESES, POSSESSIVE, NOUN_MOD, APPOSITION, MORPHOLOGY, CITATION*, and *TABLE*. The PARENTHESES signal is indicated in many examples in table 1. Examples 24 and 25 highlight portions of words that justify the POSSESSIVE and MORPHOLOGY grammatical features. CITATION refers to cases where the parenthetical relation between the citation and a constituent itself is an indicator of a relation, e.g., Example 11. APPOSITION (Examples 9) is where two NPs are abutted next to each other, typically joined by a comma or colon, such that the NPs are an IS-A relation (e.g., EXEMPLIFY or ABBREVIATE). TABLE refers to when the relation can be read off of tabular information, e.g., ABBREVIATE relations encoded in the *List of Abbreviations* section.

The other defining feature of a relation are its arguments. Our arguments include the following types of entities: (1)

documents, instantiated as citations and *self citations* (self-references like *we*, *this document* and *this research*), (2) organizations; (3) people and (4) instances of terminology (aka *jargon* or *term*), a noun or noun group that is more characteristic of technical language than of other English genres. Large projects like ACE annotate all of the entities of a particular class as well as relations between them. In our project, we only marked those entities that played the role of an argument in some relation, e.g., the instance of **melarsoprol** in example 27 was marked since it is the argument of an OPINION relation. However, if the same term appears in another context such that it is not the argument of a relation, it will not be annotated. While this restriction was motivated by resource limitations, it did not turn out to be a major stumbling block. Automatic NE extraction was used during NE processing to mark people, organizations and urls; citations are pre-marked in the PubMed corpus; and most self-citations were identified using regular expression based patterns.

Correctly identifying jargon was one of the more difficult parts of the task. Terminology or jargon notionally refers to language that is specific to a field of interest, whether that be juggling (*Mill's Mess*, *cascade pattern*) or biomedical texts (*microarray*, *bone morphogenic protein*). When choosing text to annotate as terminology, human annotators are faced with the question: "how specialized must a term be for it to be considered jargon?" To help answer this question, we have instituted a number of heuristics: (a) Would a naive adult (Homer Simpson) be familiar with this meaning of this term? If not, it is jargon; (b) if a term is found in the Juvenile Fiction sub-corpus of the Corpus of Contemporary American English[3] it is probably not jargon; among others. Given these difficulties, limiting annotation of jargon to arguments of relations adds an important filter.

4. Evaluation

While most of our articles were manually annotated a single time, we periodically annotated an article multiple times for testing purposes. The annotation was merged automatically, then corrected and augmented by an annotation supervisor. The original annotated files are then scored using the adjudicated file as an answer key. Table 2 lists the frequency of each relation in the last two adjudication exercises, along with the average precision, recall and f-measure for three annotators. We provide scores using both "strict" and "sloppy" matching criteria. A relation is "strictly" correct, if the same spans of text are chosen as arguments, and a relation is "sloppily" correct if argument spans overlap. It turns out that selecting the correct extent for terms is one of the main sources of error. Thus differences between these scores represent differences in term extent and is greatest for relations with primarily term arguments. The relation ABBREVIATE includes both acronym style abbreviations and instances of aliases (a symmetric relation between alternative names for the same thing).

There are several evaluation metrics that are commonly used for annotation. Many projects use inter-annotator agreement as a primary measure. Underlying that approach is the idea that there may not be any clearly discernible "correct" way to annotate, but if several independent annotators could at least agree on how most instances should be annotated, this was evidence that a particular way of annotating is correct. Inter-annotator agreement can be an effective method of evaluation for classification tasks, tasks in which the set of items is a given and each item in the set must receive a single classification, e.g., part of speech (POS) tagging over all the (pre-selected) tokens of a text. Unfortunately, it is difficult to make inter-annotator agreement scores simultaneously sensitive to several annotation choices: (1) the choice to mark an item or set of items as annotatable; (2) if there are multiple components of an annotation (e.g., two arguments and one signal), the choice of each component separately; and (3) the classification (e.g., choosing Exemplify, rather than Abbreviate). For strict classification tasks like POS tagging, each item receives a single classification and inter-annotator agreement measures how well the annotators agree on this classification. The complexity of relation extraction has prompted us not to use inter-annotator agreement measures.

Strict Matches				
Relation	**Freq**	**Prec**	**Rec**	**F**
EXEMPLIFY	66	85%	62%	71%
ABBREVIATE	50	95%	74%	83%
ORIGINATE	41	83%	55%	65%
OPINION	130	75%	50%	59%
REL_WORK	56	84%	44%	53%
Sloppy Matches				
EXEMPLIFY	66	91%	70%	78%
ABBREVIATE	50	99%	78%	87%
ORIGINATE	41	98%	62%	74%
OPINION	130	75%	50%	59%
REL_WORK	56	84%	44%	53%

Table 2: Average Annotator Precision and Recall

For this project, we have taken the approach that it is possible to find a "correct" gold-standard annotation, by annotating multiple times, then merging and adjudicating the results. We can then compare singly annotated text to the gold-standard annotation using precision, recall and f-measure. Other annotation projects have used this approach (Boisen et al., 2000) to evaluation, particularly for tasks related to the extraction of Named Entities, Relations and Events.

Relation annotation tends to have more errors of omission than annotation disagreements. This is clearly born out by the scores in in table 2, which show consistently higher precision than recall scores. Thus many correct answers can be obtained by simply merging the results of the two annotators. Most of the cases marked by a single annotator are correct (although not all). Furthermore, most of the conflicting annotation for our corpus is due to differences relating to spans of arguments (e.g., whether left modifiers need be included in terms), rather than the correctness of the relations themselves. As discussed in section 3., we did not have the resources to annotate all terms ahead of time. Thus and only terms that are part of a relation need be anno-

[3] http://corpus.byu.edu/coca/

tated at all. Given the wide range of possible disagreements (in contrast with, for example, part of speech tagging), we do not believe that it is possible to create a useful inter-annotator agreement score. Fortunately, comparison with a gold-standard of a representative sample clearly provides an evaluation of the quality of this type of annotation. (Min and Grishman, 2012) has taken advantage of the precision bias in ACE relations to produce a system that performs better with lots of single-pass annotation, rather than sticking to more fine-tuned multi-annotated/adjudicated annotation ACE annotation. We believe that our annotation would be compatible with this approach as well.

Other advantages of the comparison with a gold-standard approach may include: the gold standard annotation subcorpus is a useful resource in its own right and this methodology ensures that this resource will grow and improve over time; and this is the same evaluation method used for system output – thus it should be straight-forward to compare the quality of a system versus the quality of a human annotator and this is a very useful measure.

5. Pre and Post-processing

As with many annotation projects, the amount of pre-processing employed for this project has increased over time. We learned more about the task as we were doing it. We used patterns observed during annotation to write automatic procedures and used recorded annotation as a model for new types of annotation. In addition, some of these routines also reflect some of our specification refinements and clarifications. It therefore turns out that some of our pre-processing routines may also be used during post-processing to make early annotation more consistent with the new annotation. In this section, we will first describe the current state of our pre-processor and then evaluate its effectiveness as both a pre-processor for future annotation and as a post-processor to normalize older annotation to be more consistent with the newest specifications.

5.1. Preprocessing with Manual Rules

We automatically recognize some relations as well as some entities that are frequent arguments of relations. We use an NE tagger to recognize organization names (A1 of ORIGINATE). Citations to articles in the PubMed corpus have been pre-annotated as part of the PubMed corpus. However, we detect most self-citations by means of regular expressions looking for words like *we*, *our*, *this document*, *this study*, etc. (86% precision, 89% recall, 87% F-measure for a 203 instance sample).

Our pre-processing of ABBREVIATE is similar to that of (Schwartz and Hearst, 2003). In addition to detecting relations, we also use this pre-processing to detect instances of jargon, as jargon is the most common argument of the abbreviate relation (organization named entities are the second most common arguments). We find abbreviate relations in PubMed text in the following two environments: (1) explicit listing of abbreviations in *List of Abbreviations* sections; and (2) the parentheses pattern: (a) the abbreviation (A2) is in the parentheses and the full term (A1) immediately precedes the parentheses; or (b) the reverse A2 is before the parentheses and A1 is inside. For the List-of-abbreviations case, we simply parse the lists by identifying and separating the delimiters (typically colons, semicolons, periods and commas): one delimiter separates the abbreviation/term pairs and the other delimiter separates the abbreviation from the term. For the parentheses pattern, we start with the assumption that the first element next to a left parenthesis may be an abbreviation and we precede backwards from the left parenthesis in an attempt to find a set of words that "match" the abbreviation. If such a sequence is found, we have identified an abbreviate relation. If not, we also try the reverse, checking to see if an abbreviation precedes the parentheses and a full form inside the parentheses "matches" the abbreviation. It turns out that a few common patterns cover most matching cases, the most obvious one being a one to one match between letters in the abbreviation and initials in the words, e.g., Eastern Standard Time (EST). However, we must also account for common variations, e.g., Hypertext Markup Language (HTML), where an additional letter is included in the abbreviation. Variations include: subsequences of words, hyphens, plural "s", missing stop list words, among others. Once we identified an abbreviated instance of jargon, we can identify the full instance and its abbreviation elsewhere by matching these strings. Our procedure works well for Examples 1 and 2, but not 3 or 4.

In technical documents, abbreviations are almost exclusively relations between alternative forms of organization names, placenames (geopolitical entities or GPEs) or jargon terms. It turns out that a gazetteer of organization names and place names, as well as a few key-word-based regexps (association, corporation, university, etc.) can mostly identify the organization and GPE cases, leaving the jargon terms. In this way, we can automatically identify some of the jargon terms used in each document, and then mark other instances of these terms throughout the document. It turns out that this methodology has a accuracy of 76–78% (for 2681 terms of output, a human annotator identified 76% as correct, 22% as incorrect and 2% unknown).

We also have manual rules for pre-processing EXEMPLIFY, ORIGINATE and instances of OPINION relations with jargon arguments. We identify potential jargon arguments (A1 of EXEMPLIFY, A2 of ORIGINATE and OPINION) by using a dictionary derived from ABBREVIATE (see above), as well as previous annotation. Potential A1s of ORIGINATE are people and organizations, detected by our NE tagger. While the A1 of EXEMPLIFY must be an instance of jargon, the A2 is basically unrestricted (leeway is given since the A2 can lie anywhere in the IS-A hierarchy, e.g., *adreneline* is a hormone, a protein, a chemical, etc.). As noted before, no A1 is marked for OPINION. It is assumed that a lexical signal and any number of stop tokens can occur between A1 and A2 in either order (depending on the signal and relation type). Thus the EXEMPLIFY patterns resemble those of (Hearst, 1992) ("A2, such as A1", "A2 including A1", etc.). Similarly, lexical signals include *obtained from* for ORIGINATE *use* for PRACTICAL and *emerging* for SIGNIFICANT. Examples 7,8,13 and 22 follow this general pattern. In addition, we allow grammatical signals to mediate between A1 and A2 in some cases. In

those cases, there are only stop words and possibly punctuation or a possessive *'s* in between the arguments, e.g., examples 10, 12, 24 and 25, all use grammatical signals.

For CONTRAST and CORROBORATE REL_WORK relations, we apply a procedure using connectives from the PDTB (Miltsakaki et al., 2004) to link citations, as follows: (1) identify two clauses: S_1 and S_2, linked by a discourse connective indicating a CONTRAST (*however*, *in contrast*) relation or a CAUSE relation (*therefore*, *if*); (2) find citations parenthetically linked to each clause; (3) Assume that the citations connected to S_1 are A1s and those connected to S_2 are A2s for REL_WORK relations. CONTRAST discourse relations are assumed to imply CONTRAST citation relations; CAUSE discourse relations are assumed to imply CORROBORATE citation relations. Examples include 16 and 15. The full contrast and corroborate system is describe in (Meyers, 2013) including the portion that is used as a pre-processor.

5.2. Preprocessing based on Previous Annotation

After annotating approximately 200 scientific articles from the PubMed corpus, we created a knowledge base (KB) consisting of relations without citation arguments. Preprocessing with the KB essentially allows annotators to mark relations across documents. Entries in the KB consist of the source file of a relation, the minimal string in which the relation occurs, and information about the relation itself (type, subtype, signal, A1 and A2 and their respective types and subtypes). The minimal string of the relation is defined as the shortest substring of the document in which the A1, A2 and signal occur. In example 21, the minimal string would be *gene proteins used*. The KB pre-processor finds instances of these minimal strings and annotates them the same as the previous instance. If there are any modifications or rejections of the KB pre-processed relations, they are recorded. If the modifications also match previous annotation, proposed modifications of these instances are recorded for subsequent resolution by a human adjudicator.

5.3. Evaluation

We ran all of our pre-processors together on previously annotated data and then adjudicated the results, treating the automatically processed data, essentially as an annotator. We used the adjudicated file as an answer key for evaluating both the the pre-processed file and the originally annotated file. The results for 18 files are provided as table 3. During adjudication we focused only on previously annotated relations and did not attempt to find additional ones. We focused our evaluation on the relationship between previous annotation and the results of the pre-processor. Therefore the relationship between the annotation results in tables 2 and 3 is not completely clear.

In the evaluation of the pre-processing output, the recall scores indicate how much is covered in pre-processing before the annotator looks at the file. Differences between strict and sloppy evaluation indicate how many cases there are in which the annotator needs to modify the spans of pre-processed annotation (they need not find these relations independently). In part, recall indicates which portion of the task the current pre-processors are designed to handle.

Pre-Processing Evaluation (Strict)				
Relation	**Freq**	**Prec**	**Rec**	**F**
EXEMPLIFY	789	63%	16%	26%
ABBREVIATE	543	96%	53%	68%
ORIGINATE	532	84%	39%	54%
OPINION	1136	69%	35%	47%
REL_WORK	181	37%	31%	34%
Pre-Processing Evaluation (Sloppy)				
EXEMPLIFY	789	68%	17%	28%
ABBREVIATE	543	99%	55%	70%
ORIGINATE	535	92%	45%	60%
OPINION	1136	69%	35%	47%
REL_WORK	181	37%	31%	34%
Annotation Evaluation (Strict)				
Relation	**Freq**	**Prec**	**Rec**	**F**
EXEMPLIFY	789	86%	93%	89%
ABBREVIATE	543	95%	96%	95%
ORIGINATE	532	91%	84%	87%
OPINION	1136	93%	82%	87%
REL_WORK	181	63%	64%	63%
Annotation Evaluation (Sloppy)				
EXEMPLIFY	789	87%	94%	91%
ABBREVIATE	543	96%	98%	97%
ORIGINATE	532	95%	88%	92%
OPINION	1136	93%	82%	87%
REL_WORK	181	63%	64%	63%

Table 3: Evaluation of Pre- and Post-Processing

For example, the REL_WORK pre-processor is currently only designed to handle relations between citations (example 16), but not between jargon terms (example 17). Precision indicates how often the annotator will need to delete relations proposed by the pre-processor. High precision case like ABBREVIATE (94%) require the least number, although even a precision of 37% (REL_WORK) can save time since it is easier to delete a relation than to annotate one anew.

The evaluation of the annotated text (the second half of the table) is an indication of how useful these pre-processing procedures are for post-processing, i.e., correcting errors and filling gaps in previous annotation. In addition, it reflects how much the annotation task has changed over time. The two largest shifts reflected were in EXEMPLIFY and REL_WORK. For EXEMPLIFY, we had originally considered noun noun constructions to indicate exemplify in a nested fashion, so that the string *recursively enumerable sets* would generate EXEMPLIFY(*recursively enumerable sets*, *enumerable sets*) and EXEMPLIFY(*enumerable sets*, *sets*). We ended up dropping this assumption for several reasons: (i) it generated many examples that could be handled automatically; (ii) it was easy to miss lots of cases; and (iii) varied interpretations of internal NP structure yielded undesirable complications. In the case of REL_WORK, the specifications had only recently been made clear about how to handle the cross-sentence discourse structures – thus the pre-processing operation is likely to find lots of missed cases in early annotation. In this way, we are using these procedures and planning to create additional ones to offset

the problem of Specification creep, harmonizing the annotation to be consistent across all documents.

6. Annotation Tool

For annotation, we use MAE (Multi-purpose Annotation Environment) (Stubbs, 2011) [4], augmented with several features. Our modified version will shortly be release under a GNU general public license (v. 3). Our additions include: 1) **String Matching:** This feature automatically detects relations based on similar previous annotation within a document. For example, if an annotator annotates *Planck's constant* once in a document the String Matching feature will propose the identical annotation of all other near identical instances of the string *Planck's constant*; (2) **Smart List Parsing:** This feature allows the annotator to annotate an entire list of entities as a single argument of a relation and parses the list into separate relations. For example, *insulin, growth hormone* in Example 8, could be marked as an A1, with *proteins* as the A2, but 2 EXEMPLIFY relations would be created, one for each of the two conjuncts; (3) **Duplicate Detection:** The automatic detection of duplicate relations is a feature that ensures that annotators do not accidentally annotate a relation twice thinking they annotated another instead; and (4) **Feedback:** Modification and deletion of relations are recorded, both to interact with the KB pre-processor and to identify possible changes that need to be made in post-processing, e.g., if a deleted relation matches a marked relation in another file.

7. Conclusions and Future Work

We have presented an account of an annotation project in a transitional stage. We described the current state of our specifications, pre/post-processor and annotation program. As we finish annotating technical documents, we have finally figured out what our specifications should look like. We are re-tooling our pre-processors to serve as post-processors. We are considering several additional techniques to further harmonize our annotation: we anticipate using hard-coded well-formedness constraints to detect errors and give the adjudicator the option of fixing these errors (for example, see (Meyers, 2008)). Some of our relations require fixed types for A1 and A2, e.g., the A1 of ORIGINATE cannot be a jargon term, or require that A1 and A2 share the same class, e.g., ABBREVIATE can be between two jargon terms, organizations, etc. Furthermore, we can use our knowledge of known issues such as instances of nested EXEMPLIFY relations (the *recursively enumerable sets* example) to create additional filters.

Subsequent to the work described here, we used the same specifications with some modifications to annotate 26 US patents, a related but different genre of documents from technical articles. We have mainly been focusing on the post-processing phase and we are working on trying to improve our annotation quality before releasing it to the public. We are considering the issue of how to produce the best resource: is it better to multiply annotate, correct and adjudicate single-pass annotation or annotation from previous versions of the specifications? Or is it better to just annotate additional documents? We believe that corrections for accuracy are worthwhile, particularly if the corrections are quick and accurate. However, corrections for improving recall might not be. For this task, annotators are more likely to miss instances than to overmark. Thus this annotation effort probably has the characteristics described in (Min and Grishman, 2012) – for high precision annotation, it may be more effective to annotate more than to multiply annotate and adjudicate. (Min and Grishman, 2012) showed that high performance relation extraction could be achieved on single-pass high-precision/low-recall annotation by using methods that anticipate that some of the correct responses are missing. These methods can reach the same level of performance with single-pass data using 10% more data than comparable systems using multi-pass annotation (saving about 2/3 the annotation cost).

We have been developing additional tools based on manual-rules to automatically annotate documents for purposes of better pre- and post-processing. These include: a system for automatically marking jargon terms using a method based on noun group chunking[5]; and we have created a manual rule-based system that find relations between these terms. Our future focus will be on refining our post-processor and using it to improve the quality of our annotation with the least amount of effort. We then plan to distribute the cleaned-up annotation under a permissive open source licence (e.g. Apache). We also intend to distribute our modified version of Stubbs' MAE annotation tool (before the workshop).

Acknowledgments

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20154. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government. Approved for Public Release; Distribution Unlimited.

8. References

A. Athar. 2011. Sentiment Analysis of Citations using Sentence Structure-Based Features. In *Proc. of the ACL 2011 Student Session*.

O. Babko-Malaya, A. Meyers, J. Pustejovsky, and M. Verhagen. 2013a. Modeling Debate within a Scientific Community. In *SOCIETY 2013*.

O. Babko-Malaya, P. Thomas, D. Hunter, A. Meyers, J. Pustejovsky, M. Verhagen, and G. Amis. 2013b. Characterizing Communities of Practice in Emerging Science and Technology Fields. In *SOCIETY 2013*.

[4] `http://code.google.com/p/mae-annotation/`

[5] We have also experimented with terminology extraction methods like those of (Navigli and Velardi, 2004). While such methods are good at finding key topic words, these word lists did not provide us with sufficient recall of relation arguments.

C. F. Baker, C. J. Fillmore, and J. B. Lowe. 1998. The Berkeley FrameNet Project. In *Coling-ACL98*, pages 86–90.

S. Boisen, M. Crystal, R. M. Schwartz, R. Stone, and R. M. Weischedel. 2000. Annotating resources for information extraction. In *LREC*.

T. U. Daim, G. Rueda, H. Martin, and P. Gerdsri. 2006. Forecasting emerging technologies: Use of bibliometrics and patent analysis. *Technological Forecasting and Social Change*, 73(8):981–1012.

Randy Allen Harris. 1995. *The linguistics wars*. Oxford Univ. Press, Oxford.

M. Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *ACL 1992*, pages 539–545.

A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman. 2004. Annotating Noun Argument Structure for NomBank. In *Proceedings of LREC-2004*, Lisbon, Portugal.

A. Meyers. 2008. Using treebank, dictionaries and glarf to improve nombank annotation. In *Proceedings of The Linguistic Annotation Workshop, LREC 2008*, Marrakesh, Morocco.

A. Meyers. 2013. Contrasting and Corroborating Citations in Journal Articles. In *RANLP-2013*.

E. Miltsakaki, A. Joshi, R. Prasad, and B. Webber. 2004. Annotating discourse connectives and their arguments. In A. Meyers, editor, *NAACL/HLT 2004 Workshop: Frontiers in Corpus Annotation*, pages 9–16, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

B. Min and R. Grishman. 2012. Compensating for Annotation Errors in Training a Relation Extractor. In *EACL-2012*.

R. Navigli and P. Velardi. 2004. Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. *Computational Linguistics*, 30.

M. Palmer, D. Gildea, and P. Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

A. Schwartz and M. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Pacific Composium on Biocomputing*.

A. Stubbs. 2011. MAE and MAI: Lightweight Annotation and Adjudication Tools. In *Proceedings of the Linguistic Annotation Workshop V*.

S. Teufel, A. Siddharthan, and C. Batchelor. 2009. Towards domain-independent argumentative zoning: Evidence from chemistry and computational linguistics. In *EMNLP-2009*.

P. Thomas, O. Babko-Malaya, D. Hunter, A. Meyers, and M. Verhagen. 2013. Identifying Emerging Research Fields with Practical Applications via Analysis of Scientific and Technical Documents. In *ISSI 2013*.